

# Partitioning Strategies and Task Allocation for Target-tracking with Multiple Guards in Polygonal Environments

Hamid Emadi<sup>1</sup>, Tianshuang Gao<sup>2</sup> and Sourabh Bhattacharya<sup>3</sup>

**Abstract**—This paper presents an algorithm to deploy a team of *free* guards equipped with omni-directional cameras for tracking a bounded speed intruder inside a simply-connected polygonal environment. The proposed algorithm partitions the environment into smaller polygons, and assigns a guard to each partition so that the intruder is visible to at least one guard at all times. Based on the concept of *dynamic zones* introduced in this paper, we propose event-triggered strategies for the guards to track the intruder. We show that the number of guards deployed by the algorithm for tracking is strictly less than  $\lfloor \frac{n}{3} \rfloor$  which is sufficient and sometimes necessary for coverage. We derive an upper bound on the speed of the mobile guard required for successful tracking which depends on the intruder's speed, the road map of the mobile guards, and geometry of the environment. Finally, we extend the aforementioned analysis to orthogonal polygons, and show that the upper bound on the number of guards deployed for tracking is strictly less than  $\lfloor \frac{n}{4} \rfloor$  which is sufficient and sometimes necessary for the coverage problem.

## I. INTRODUCTION

In the past decade, mobile robotic networks have become a ubiquitous part of human society [1], [2], [3], [4]. An important application has been in the area of surveillance and data gathering. Although, electronic and biometric techniques are emerging rapidly in security applications, vision-based monitoring using static camera networks is still the most prevalent technique used for persistent surveillance [5]. However, the scalability of this technique is poor due to the huge amount of data acquired by modern camera networks. Mobile camera networks alleviate the problem of *data deluge* to a significant extent. In this work, we explore a scenario in which a team of mobile agents that can visually track entities in the environment are deployed in a surreptitious manner for tracking a mobile intruder.

Target tracking refers to the problem of tracking a mobile object, called a *target*. Based on the sensing modality and sensing constraints, there is a range of problems that can be addressed under this category. Several variants of the target-tracking problem have been considered in the past that consider constraints in motion as well as sensing constraints for both agents. For an extensive discussion regarding the previous work on target tracking and its applications, we refer the reader to [6], [7].

In an adversarial setting, the target-tracking problem gives rise to a visibility-based pursuit evasion game [8]. Tools from differential game theory have been used to investigate

motion and control strategies for the observer. However, the aforementioned tools fail to elucidate the dependence of the tracking performance on the geometric complexity of the environment. In [9], the authors capture this relationship by drawing an explicit connection between target-tracking and art-gallery problems involving mobile guards [10]. This paper is in a similar vein, and investigates partitioning techniques for a polygon to deploy guards for tracking a mobile intruder.

The art gallery problem is a classic problem in which the goal is to deploy the minimum number of guards in a polygonal environment to ensure visual coverage of the entire environment. Generally, the guards in the art gallery problem are assumed to be *point guards* which are stationary agents equipped with omni-directional camera having infinite sensing range. Although, the art gallery problem for several classes of polygons is NP-hard [11], there are tight bounds on the number of stationary guards that can cover a simple  $n$ -sided polygon and non-simple polygons containing holes [12], [10], [13]. The notion of mobile guards was introduced in [14]. Mobile guards are classified as edge, diagonal or free guards, depending on the path which they are patrolling. Similar to the point guards, there are tight upper and lower bounds on the number of diagonal and edge guards required to cover a polygon [15], [10].

Beside art-gallery problems, there have been several works that have addressed the coverage problem in a polygon. In [16], the authors study the problem of guarding a polygon under the  $\Delta$ -guarding constraint. This constraint ensures that all sides of a convex object are visible in the environment at all times. The problem of inspecting an entire polygon with a group of mobile guards is called the *watchman's route* problem. In [17], [18], two variations of the watchman's route problem, one in which the guards travel on minimum-length route and the other in which the minimum number of mobile guards are deployed, are examined. These algorithms provide a practical method to search an intruder inside a polygon. Authors in [19] propose a motion-planning algorithm for a group of sliding robots, assuming that they move along the pre-located line segments with a constant speed to detect all the evaders with unbounded speed. Such problems are classified as *search* problems [1]. A visibility based search in a polygon wherein all guards and intruders have bounded speeds is examined in [20]. In [21], the authors present a rotation schedule for a group of static searchlight sensors in order to detect all targets in the polygon. Furthermore, distributed algorithms are proposed in [22], [23] for a group of mobile guards to reach points from which the entire polygon

<sup>1</sup>Hamid Emadi and Sourabh Bhattacharya are with the Department of Mechanical Engineering, Tianshuang Gao is with the Department of Computer Science, Iowa State University, Ames, IA 50011 {emadi, sbhattach, tsgao}@iastate.edu

is visible. Using these algorithms, a group of mobile sensors can fully cover the entire polygon from an arbitrary initial position. For an elaborate survey on art gallery problems and coverage algorithms for polygons, we refer the reader to [24], [10].

Another variant of the surveillance problem involves covering the entire environment with pan, tilt, zoom (PTZ) cameras for tracking or detecting an intruder. The challenge here is to plan and coordinate the motion of cameras to track all the evaders over long time intervals. Authors in [3], [5] provide a dynamic control over a network of PTZ camera in order to track an evader with a random trajectory. Some of the tracking algorithms are based on the current position of the target. For instance, a collision free motion strategy is proposed in [4] for a differential drive mobile robot for tracking a target. On the other hand, [25], [2] consider a scenario in which the sensor has some apriori knowledge about the target's behavior, and predicts its new position for tracking tasks.

In this work, we formulate the tracking problem as a task allocation problem. The guards are assigned the task of tracking the intruder in a partition of the polygon. The contributions of this paper are as follows: (i) We propose a deployment algorithm to track the intruder which requires less than  $\lfloor \frac{n}{3} \rfloor$  guards; the sufficient and sometimes necessary number of guards for coverage. This provides a new upper bound on the number of guards required for target tracking. (ii) We introduce the idea of "dynamic zones", and use it to propose event-triggered strategies for the guard to track the intruder. (iii) We provide an upper bound on the speed of the guard required to track the intruder. This bound encompasses the geometric parameters of the environment. (iv) We extend the techniques to orthogonal polygons to propose deployment strategy for  $\lfloor \frac{n}{4} \rfloor$  guards to track successfully.

The rest of the paper is organized as follows. In section II, we present the problem statement. In section III, we propose a deployment of guards in a general polygon by partitioning it into basic polygons. In section IV, the deployment and strategy of guards in the basic polygons are described. In section V, we focus on a special class of polygons, which is called orthogonal polygons. In Section VI, we conclude the paper with a few remarks.

## II. PROBLEM STATEMENT

Consider a team of mobile agents, called *guards*, inside a simply-connected polygonal environment  $P$  with  $n$ -sides. Assume that each observer is equipped with an omni-directional camera having an infinite sensing range. The environment contains another mobile agent called the *intruder*. The objective of the guards is to track the intruder inside the polygon at all times. The speed of the guard and intruder are denoted as  $v_e \in [0, \bar{v}_e]$  and  $v_p \in [0, \bar{v}_p]$ , respectively. Two points inside the polygon are mutually *visible* if the line segment joining them, called the *line of sight* (LOS), is contained in  $P$ . Since we consider infinite sensing range, LOS can be obstructed only by obstacles (i.e. the reflex corners of the environment).

According to the art gallery theorems,  $\lfloor \frac{n}{3} \rfloor$  static guards are sometimes necessary and always sufficient to cover the interior of a simply connected polygon  $P$  of  $n$ -vertices. In a coverage problem, the objective is to cover the entire environment simultaneously, regardless of the intruder's position. There are some scenarios in which certain intruders in the environment should be tracked. It would be extravagant for these scenarios to cover the entire environment. Contrary to the coverage problem, it is sufficient to keep the intruder in the guards' field of view in the tracking problem. Since the intruder has bounded speed, deployment of the mobile guards with bounded speed is practical. In this work, we deploy a group of mobile and static guards to track the intruder within a polygon  $P$ , such that the total number of required guards is strictly less than  $\lfloor \frac{n}{3} \rfloor$ . The reduction of the number of guards comes from the motion ability of the guards. In [9], the authors presented a deployment for diagonal guards based on the triangulation of the polygon. In this work, we investigate deployment strategies for *free guards*, i.e., guards that are free to move inside the polygon.

## III. TARGET TRACKING IN GENERAL POLYGONS

In this section, we present a technique to partition a general polygon into smaller polygonal regions so that guards can be allocated to each region for tracking the intruder. This can be modeled as a multi-robot task allocation problem (MTAP) wherein guarding the different regions are tasks that need to be allocated to the robots. This can also be modeled as a resource allocation problem wherein guards are resources that need to be allocated to the different regions. The goal of task allocation is to assign robots to subtasks in order to reach the performance of the system. The robots share their individual expected task to maximize the team performance. Algorithms to solve the matching problem for weighted bipartite multi-graphs are used to solve MTAP [26]. There is a mathematical modeling and analysis of the collective behavior of dynamic task allocation in [27]. In this work, we assign a guard to a partition and the motion strategy of the mobile guard is given based on the dynamics of the intruder.

The following lemma presents the main idea of the partitioning technique:

*Lemma 1:* [10] Consider a polygon  $P$  with  $n \geq 10$  vertices, and  $T$  a triangulation graph of  $P$ . There exists a diagonal  $d$  in  $T$  that divides  $T$  into two partitions, one of which contains  $k = 5, 6, 7$ , or  $8$  arcs corresponding to edges of  $P$ .

From lemma 1, we can recursively partition a polygon  $P$  into smaller polygons such that each partition contains 6, 7, 8, or 9 edges. The process terminates with a polygon with less than 6 edges.

*Definition 3.1:* *Minimal partitioning* is partitioning a polygon  $P$  using diagonals of triangulation  $T$  into smaller partitions such that a pair of partitions share at most one edge and except one partition, all partitions contain 6, 7, 8, or 9 edges. The remaining partition may be a pentagon, quadrilateral or triangle.

First, we show that the sufficient number of mobile guards to track a target is, in general, less than the number of static guards required to cover the entire polygon.

*Proposition 1:* Every polygon  $P$ , can be guarded by less than  $\lfloor \frac{n}{3} \rfloor$  mobile guards if following statements hold:

- 1) Minimal partitioning  $P$  contains at least 3 partitions.
- 2) For every partition  $P_i$  the sufficient number of mobile guards to track an intruder is less than  $\lfloor \frac{n_i}{3} \rfloor$ .

*Proof:* Consider a triangulation  $T$  of polygon  $P$ . According to lemma 1, any general  $n$ -sided polygon  $P$  can be partitioned into  $P_i$ 's, such that each partition is a hexagon, septagon, octagon or nonagon. Furthermore, it may contain only one partition  $P'$  which is a triangle, quadrilateral or pentagon. Let  $r$  denotes the number of  $P_i$ 's. Let  $\hat{k}_i + k_i$  be the number of edges of partitions, where  $\hat{k}_i \in \{1, 2\}$ , and  $k_i \in \{4, 5, 6, 7\}$ . Partitioning  $P$  may result in one more partition  $P'$  with  $k'$  edges, where  $k' \in \{0, 3, 4, 5\}$ . Since the partitions are obtained from the original polygon, the following relation holds between  $n, k_i, \hat{k}_i$  and  $k'$ ,

$$n + 2(r - 1) - k' = \sum_{i=1}^r k_i + \hat{k}_i. \quad (1)$$

We have assumed that each  $P_i$  can be guarded by at least one guard less than the required number of static guards, that is  $\lfloor \frac{k_i + \hat{k}_i}{3} \rfloor - 1$ . We need to show that the total number of guards is less than  $\lfloor \frac{n}{3} \rfloor$ . For those partitions, in which  $P'$  exists (i.e.  $k' \neq 0$ ) one more guard is needed to cover  $P'$ . Therefore, the total number of guards is given by the following:

$$\begin{aligned} & \sum_{i=1}^r \lfloor \frac{k_i + \hat{k}_i}{3} \rfloor - r + 1 \leq \lfloor \frac{\sum_{i=1}^r k_i + \hat{k}_i}{3} \rfloor - r + 1 \\ & \leq \lfloor \frac{n + 2(r - 1) - k' + 2}{3} \rfloor - r + 1 \\ & \leq \frac{n - k' + 2(r - 1) + 2}{3} - r + 1 = \frac{n - k' - r + 3}{3} \\ & < \lfloor \frac{n}{3} \rfloor \quad \text{for } r \geq 3. \end{aligned} \quad (2)$$

For those polygons, in which  $P'$  does not exist (i.e.  $k' = 0$ ), total number of guards is given by the following:

$$\begin{aligned} & \sum_{i=1}^r \lfloor \frac{k_i + \hat{k}_i}{3} \rfloor - r \leq \lfloor \frac{\sum_{i=1}^r k_i + \hat{k}_i}{3} \rfloor - r \\ & \leq \lfloor \frac{n + 2(r - 1)}{3} \rfloor - r \leq \frac{n + 2(r - 1)}{3} - r = \frac{n - r - 2}{3} \\ & < \lfloor \frac{n}{3} \rfloor \quad \text{for all } r. \end{aligned} \quad (3)$$

#### IV. TRACKING IN BASIC POLYGONS

In this section, we present deployment strategies for mobile guards within each partition.

According to proposition 1, as the number of partitions increases, the sufficient number of guards is decreased.

We propose motion strategy, and derive the maximum speed required for the mobile guards to track an intruder

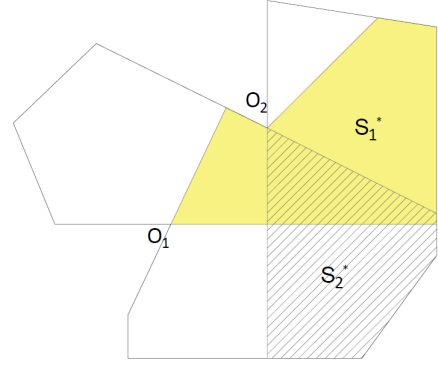


Fig. 1: Star regions in a polygon containing two reflex vertices.  $S_1^*$  yellow region, and  $S_2^*$  hatched region.

within each  $P_i$  such that the second condition in proposition 1 is satisfied. Let  $S_i^*$  denote the *star region* which is defined as the bounded area, inside the environment, across the reflex vertex  $O_i$ . In other words,  $S_i^*$  is the set of all points in  $P$  such that the connecting line between those points and  $O_i$  lies in  $P$ , and lies in the area constructed by two edges of reflex vertex  $O_i$ . Figure 1 shows the star region in a polygon. In the proposed technique, the motion strategy of the guards are based on the number of disjoint star regions in  $P_i$ . The following propositions characterizes the maximum number of disjoint star regions in hexagons, septagons and octagons.

*Lemma 2:* Every polygon  $P$  with a reflex vertex and  $r \geq 2$  non-intersecting edges with the star region has at least  $r + 3$  edges. Moreover,  $P$  contains at least 5 edges for  $r = 1$ .

*Proof:* Any reflex vertex contains two intersecting line segments. Extending these line segments separates the entire space into four quadrants, where the star region is the first quadrant, and numbering proceeds counterclockwise. All edges should lie either in quadrant one, two or four. For  $r > 1$ , two edges are used to construct the reflex vertex, and consequently they have intersection with the star region,  $r$  edges may be on quadrants two or four, but one more edge must intersect with the star region because of the simply connectedness of the polygon. Therefore, the polygon contains  $r + 3$  edges. For  $r = 1$ , two edges are needed to connect the non-intersecting edge to the other edge of the reflex corner. As a consequence,  $P$  contains at least 5 edges. ■

*Lemma 3:* In a pentagon with one reflex vertex and two non-intersecting edges with the star region, sum of each pair of angles of pentagon on each side of star region must be less than  $\pi$  (i.e.  $\alpha_1 + \alpha_2 < \pi$ ).

*Proof:* Figure 2a shows a pentagon with one reflex vertex and two edges that do not intersect with the star region ( $AB$  and  $CD$ ). Extending two edges corresponding to the reflex vertex, separates the entire space into four regions, one of which contains the star region.  $OA$  and  $OD$  are used to draw a reflex vertex, and other edges should be on second and fourth quadrants, while the first quadrant contains the star region. Therefore, without loss of generality, every pentagon containing one non-intersecting edge with star region, can

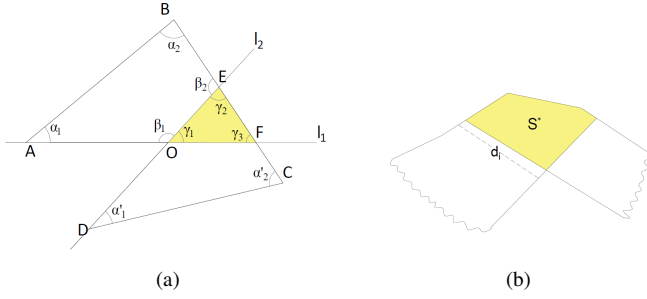


Fig. 2: (a) A pentagon with one reflex vertex and two non-intersecting edges with the star region. (b) Line segment  $d_i$  to partition a polygon  $P_n$ .

be considered as Figure 2a. From Figure 2a,

$$\begin{aligned}\alpha_1 + \alpha_2 + \beta_1 + \beta_2 &= 2\pi, \\ \alpha_1 + \alpha_2 + \gamma_1 + \gamma_2 + \gamma_3 + \gamma_3 &= 2\pi.\end{aligned}$$

Since  $\gamma_1 + \gamma_2 + \gamma_3 = \pi$  and  $\gamma_3 > 0$ ,  $\alpha_1 + \alpha_2 < \pi$ . The same argument holds for  $\alpha'_1 + \alpha'_2 < \pi$ . ■

**Proposition 2:** Every polygon containing 2, 3, 4 or 5 disjoint star regions must have at least 6, 7, 9 or 10 edges, respectively.

*Proof:* We consider a  $n$ -sided polygon  $P$ , and let  $n_d \in \{2, 3, 4, 5\}$  be the number of disjoint star regions. Each star region is a convex polygon with two edges inside  $P$ , and only one common vertex with reflex vertices of  $P$  and some parts of the boundary of  $P$  (yellow region in Figure 2b). Based on the definition of a star region, we can find line segments  $d_1, \dots, d_{n_d-1}$  which partition  $P$  into  $n_d$  parts, and each part contains one star region. Figure 2b shows a plausible choice of  $d_1$ . Now, let  $G(v, e)$  be a dual graph for  $P$  such that each partition with only one star region is a node in  $G$ . An edge exists between a pair of nodes if their corresponding partitions are separated by a common  $d_i$ . Since  $P$  is simply connected, graph  $G$  is connected. Moreover  $G$  has  $n_d$  nodes and  $n_d - 1$  edges. Therefore,  $G$  is a tree. All possible non-isomorphic trees with  $n_d \in \{2, 3, 4, 5\}$  nodes are illustrated in Figure 3.

Each node with degree one corresponds to a polygon with a reflex corner and a non-intersecting edge with star region. As a result of lemma 2, each node with degree one must be a polygon which contains at least 5 edges. Additionally, nodes with degree 3 and 4 must be hexagon and septagon, respectively. Based on lemma 3, two consecutive nodes with degree two cannot both be a pentagon, which implies one of the partitions must contain at least 6 edges. In order to find the total number of edges of  $P$ , we have to exclude the common edges in partitions (i.e.  $d_i$ 's). Moreover the edges belonging to  $P$  which intersect  $d_i$ 's should be counted only once. Computing the total number of edges for the graphs for  $n_d = 2, 3, 4$  and 5, leads to  $n = 6, 7, 9$  and 10, respectively. Therefore, a hexagon, a septagon and an octagon can at most contain 2, 3 and 3 disjoint star regions, respectively. ■

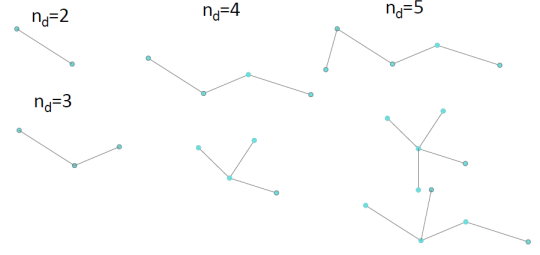


Fig. 3: Dual graph of  $P$  for  $n_d = 2, 3, 4, 5$ .

### A. Tracking in a Hexagon

In this section, we show that every hexagonal environment can be guarded by a mobile guard if the maximum speed of guard is greater than or equal to a threshold, which depends on the geometry of the environment and maximum speed of evader (i.e.  $\bar{v}_p \geq v_p^*$ ). First, we propose an initial deployment of the pursuer for different initial positions of the evader based on the number of disjoint star regions in the environment, and then we present a strategy for the pursuer's motion. Based on the number of reflex vertices, every hexagon can be classified into the following categories:

- Case I: Convex hexagon (no reflex vertex).
- Case II: Hexagon with one reflex vertex.
- Case III: Hexagon with two reflex vertices.
- Case IV: Hexagon with three reflex vertices.

In Case I, a static guard can cover the entire hexagon since it is convex. In case II, deploying only one static guard in  $S^*$  can cover the entire hexagon.

In case III, let  $\gamma_p$  denotes the smallest path within the hexagon which connects  $S_1^*$  and  $S_2^*$ .  $S_1^* \cap S_2^* \neq \emptyset$  implies  $|\gamma_p| = 0$ , and there exists at least one point in  $S_1^* \cap S_2^*$  which can cover the entire environment. Deploying one static guard in  $S_1^* \cap S_2^*$  would be enough to cover the hexagon. When  $|\gamma_p| \neq 0$ , the environment can be partitioned into three parts, which determines the strategy of mobile guard. This partitioning is based on defining a *dynamic zone* denoted by  $D_i$  around a reflex vertex  $C_i$ .

Refer to Figure 4a. Assume the pursuer lies on point  $H_0$ . Let  $l_i$  be the ray that begins from vertex  $C_i$  and it lies in the area of visibility of  $H_0$  (i.e.  $l_i \in \mathcal{V}(H_0)$ ). For any line segment  $d_i := H_0H_1$  which connects  $H_0$  to  $S_i^*$ , dynamic zone is defined as the set of all points in the environment whose distance from  $l_1$  is less than  $r_i = \frac{\bar{v}_e}{\bar{v}_p} d_i$ . The green area in Figure 4a shows the dynamic zone. In general,  $D_i$  encompasses a sector of a disc with radius of  $r_i$ , and the area between two parallel lines, one of which tangents to the disc and the other one passes through  $C_i$ . When the evader enters region  $D_i$ , the pursuer initiates motion from  $H_0$  towards  $H_1$  along the path  $d_i$ . The boundary of  $D_i$  contains two main parts,  $\partial D_i$  and  $l_i$  belong to the free space.  $\partial D_i$  and  $l_i$  correspond to the points  $H_0$  and  $H_1$ , which implies that if the evader enters  $D_i$  from  $\partial D_i$  or  $l_i$  then the pursuer should lie on  $H_0$  and  $H_1$ , respectively. Based on the distance of the evader with respect to  $\partial D_i$  and  $l_i$ , the pursuer should

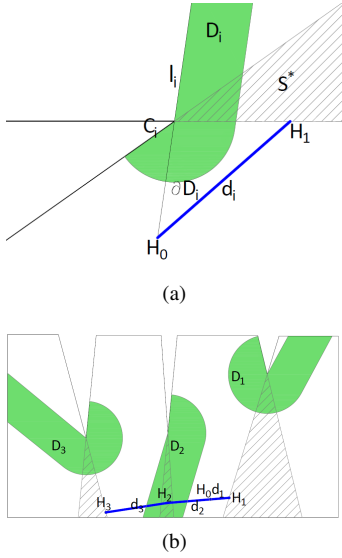


Fig. 4: (a) Dynamic zone around corner  $C_i$ , (b) Dynamic zones and road map among star regions

be on a certain position on  $d_i$ . In other words, there exists an isomorphic mapping from  $D_i$  to  $d_i$ .

We can extend the concept of dynamic zone to the environments containing multiple reflex corners. First, we construct a road map which connects all star regions. Figure 4b shows a road map which connects three star regions. Next, based on the road map we construct the dynamic zones around each corner. The dynamic zones for a certain road map should satisfy the following conditions:

- Let  $D_i$  and  $D_j$  be two dynamic zones which are separated by a static zone.  $D_i$  and  $D_j$  should be consistent. By consistent we mean, two boundaries of  $D_i$  and  $D_j$  which are connected to the same static zone should correspond to the same point according to the pursuer's position.
- Every two dynamic zones cannot contain common points except some points on their boundaries.
- Every  $D_i$  cannot contain a reflex vertex except  $C_i$ .

When a hexagon contains two disjoint star regions, the maximum radius of dynamic zone is the distance between two reflex vertices  $o_{12}$ . Therefore, the minimum  $\bar{v}_p$  is  $\frac{|\gamma_p|}{r_{12}} \bar{v}_e$ .

In case IV, the environment has three reflex corners. If  $S_1^* \cap S_2^* \cap S_3^* \neq \emptyset$ , the entire hexagon can be covered by only one static guard. For environments in which  $S_1^* \cap S_2^* \cap S_3^* = \emptyset$  (Figure 5c),  $\gamma_p$  is defined as the shortest path which connects the intersection set of two star regions. That is the shortest path between  $S_i^* \cap S_j^*$  for  $i \neq j, i, j = 1, 2, 3$ . Given  $\gamma_p$ , the deployment and motion strategy of the pursuer can be determined in the same way as described for case III. The following proposition summarizes the results in this subsection.

**Proposition 3:** A mobile guard can track an evader in a hexagonal environment for infinite time if the following

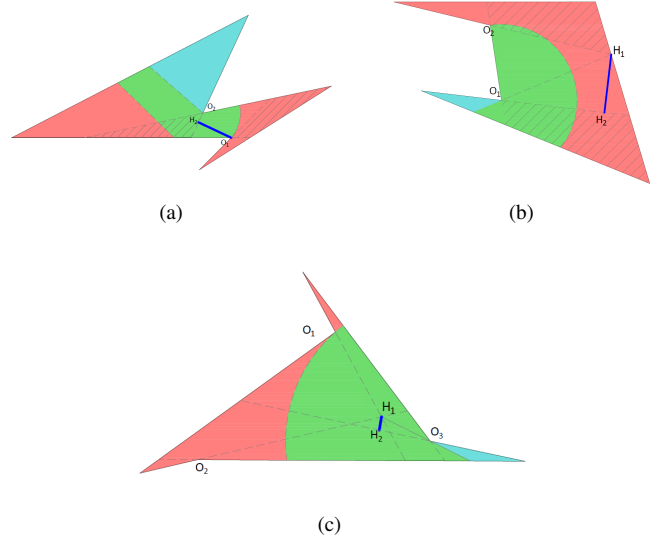


Fig. 5: Hexagonal environment (a), (b) Case III, (c) Case IV. Evader's position in every colored partition, determines specific deployment and strategy of pursuer.

condition is satisfied:

$$\bar{v}_p \geq v_p^* \quad (4)$$

$$v_p^* = \frac{|\gamma_p|}{r} \bar{v}_e, \quad (5)$$

where  $\gamma_p$  is the shortest path which connects all star regions, and  $r$  is the radius of arc segment in  $D_i$ .

### B. Tracking in a Septagon

In this subsection, we analyze the deployment and motion strategy for a single guard in a septagonal environment. Similar to the hexagonal environments, we can classify septagons according to the number of reflex corners, which is at most five.

- Case I: Convex septagons, which can be covered with one static guard.
- Case II: Septagons containing only one reflex vertex, which can be covered with only one static guard which is placed in  $S^*$ .
- Case III: Septagons containing two reflex vertices. When star regions are completely disjoint, a mobile guard with a maximum speed, given by proposition 3 is needed. Otherwise one static guard is enough to cover the entire environment.
- Case IV: Septagonal environments in which three reflex corners exist. If intersection of all star regions is not empty set (i.e.  $S_1^* \cap S_2^* \cap S_3^* \neq \emptyset$ ), then deployment of a static guard in  $S_1^* \cap S_2^* \cap S_3^*$  is enough to cover the entire environment. Now consider the case in which two star regions intersect,  $S_1^* \cap S_2^* \neq \emptyset$ , but third one ( $S_3^*$ ) is disjoint (i.e.  $S_1^* \cap S_3^* = \emptyset, S_2^* \cap S_3^* = \emptyset$ ). In such cases, a mobile guard on the shortest path between  $S_1^* \cap S_2^*$  and  $S_3^*$  can track an evader for infinite time.



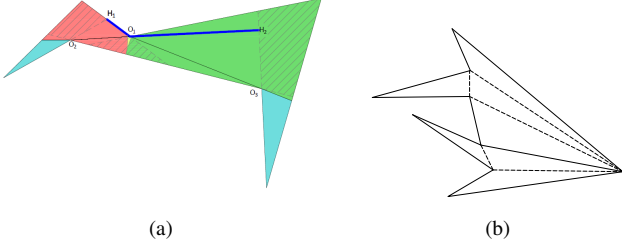


Fig. 6: (a) Septagonal environment, (b) Partitioning 9-gon into hexagon and pentagon

When all three star regions are disjoint (i.e.  $S_i^* \cap S_j^* = \emptyset, i \neq j, i, j : 1, 2, 3$ ), one mobile guard can track an evader. In this case, we partition the environment based on the motion of pursuer on the shortest path, which connects the furthest star regions. Figure 6a shows a septagon and three disjoint star regions. The pursuer's trajectory contains two paths  $H_1O_1$  and  $O_1H_2$ . In this case  $v_p^*$  is given by the following expression:

$$v_p^* = \min_{r_1, r_2} \max \left\{ \frac{d_1 \bar{v}_e}{r_1}, \frac{d_2 \bar{v}_e}{r_2} \right\}, \quad (6)$$

where  $d_1$  and  $d_2$  are lengths of  $H_1O_1$  and  $O_1H_2$ , respectively.  $r_1$  and  $r_2$  are the radius of sector of dynamic zones around  $O_1$  and  $O_2$ , respectively. Note that  $r_1$  and  $r_2$  are subject to the following constraints:

$$r_1 \leq o_{12}, \quad r_2 \leq o_{13}, \quad r_1 + r_2 \leq o_{23}. \quad (7)$$

Since  $\frac{d_1 \bar{v}_e}{r_1}, \frac{d_2 \bar{v}_e}{r_2}$  are decreasing function of  $r_1, r_2$ ,  $v_p^*$  occurs at the boundary of one of the constraints.

In all other cases, we can construct a road map among the intersection of star regions and disjoint star regions. The lower bound for  $\bar{v}_p$  is  $\min \max \left\{ \frac{d_i}{r_i} \bar{v}_e \right\}$ .

According to the proposition 2, octagons have at most three disjoint star regions. Hence, all the above cases hold for general octagonal environments.

*Deployment of a guard in a hexagon, septagon and octagon:* As mentioned earlier, only one guard is enough to track an evader within a hexagon, a septagon or an octagon. To deploy a guard, we need to identify all star regions in these polygons, and subsequently partition the polygon based on the number of star regions and maximum speed of evader. Deployment of the guard depends on the position of evader with respect to the partitions. Moreover,  $v_p^*$  is computed based on  $\bar{v}_e$  and geometry of the star regions. As the evader moves inside the polygon, the guard should adopt an appropriate strategy with respect to the evader's position in the partitions.

### C. Tracking in a Nonagon

In this subsection, we show that two guards, one static and one mobile guard, is enough to track an evader in a 9-gon. In every triangulation of a 9-gon, we can find a diagonal such that it partitions the environment into two parts, one of which is a pentagon and the other one is a hexagon. Deploying a

static guard is enough to cover the pentagon, and a mobile guard can track the evader within the hexagon for infinite time. Hence, two guards are sufficient to guard a 9-gon.

Now we have shown that hexagons, septagons, octagons and nonagons can be guarded by static or mobile guards such that the number of required guards is strictly less than the number of required static guards to cover these environments. Therefore, according to proposition 1, any general polygon can be guarded with group of  $n_g$  static and mobile guards such that  $n_g < \lfloor \frac{n}{3} \rfloor$ . The procedure of deploying guards is based on the minimal partitioning of polygon  $P$  into hexagons, septagons, octagons and nonagons. Each partition is guarded by one or two guards based on the number of reflex vertices and the relative placement of star regions. Algorithm 1 presents the procedure of deploying guards in a general polygon. Figure 7 shows minimal partitioning of a 20-gon and deployment of guards for infinite time tracking of an evader. Partitioning and motion strategy of mobile guards, based on the intruder's trajectory, are shown in our video submission.

---

**Algorithm 1** Infinite time tracking of the intruder in a general polygon.

---

```

1: procedure DEPLOYING GUARDS ( $P, e_0$ )
2: Input:  $P$  is a polygon,  $e_0$  is the initial position of the intruder.
3:   Minimal partitioning  $P$  to  $\mathcal{P} = \{P_1, \dots, P_r\}$ 
4:    $v_p^* \leftarrow 0$ 
5:   for any  $P_i \in \mathcal{P}$  do
6:     if  $P_i$  is hexagon, septagon or octagon, then
7:       Follow the deployment of a guard in a hexagon, septagon or octagon.
8:     else
9:       Follow the deployment of guards in a nonagon environment.
10:    end if
11:    Compute  $v_{p_i}^*$ .
12:     $v_p^* \leftarrow \max(v_{p_i}^*, v_p^*)$ 
13:  end for
14: end procedure

```

---

## V. TARGET TRACKING IN ORTHOGONAL POLYGONS

In this section, we present deployment strategies for guards in orthogonal polygons, which is an important subclass of polygons. An orthogonal polygon is one whose edges are all parallel to the axes of Cartesian coordinate. Therefore, all the angles of an orthogonal polygon are either  $90^\circ$  or  $270^\circ$ . We take all edges of  $P$  to be horizontal and vertical without loss of generality. We begin with the following lemmas.

*Lemma 4:* [10] Every orthogonal polygon  $P$  is convexly quadrilateralizable.

Given the quadrilateralization of an orthogonal polygon, we can associate a graph similar to the dual graph of the triangulation of a polygon. Let  $G$  be a graph such that every vertex corresponds to a quadrilateral in quadrilateralization

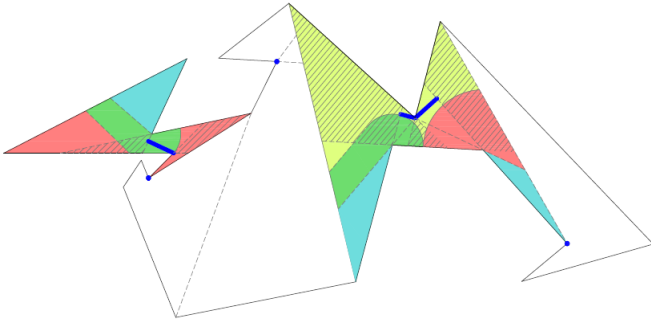


Fig. 7: Deployment of guards in a 20-gon. Solid blue lines show the paths of mobile guards and blue dots are static guards.

$Q$  of  $P$ , and two vertices share an edge if their corresponding quadrilaterals share an edge.

**Lemma 5:** [10] For every quadrilateralization  $Q$  of an orthogonal polygon  $P$ , the dual graph  $G$  is a tree with each node of maximum degree 4.

**Lemma 6:** The quadrilateralization  $Q$  of  $P$  can be partitioned by diagonals of  $P$  into smaller partitions,  $P_i$ 's, such that each  $P_i$  contains 2, 3, or 4 quadrilaterals. Furthermore, there may be one remaining quadrilateral, denoted by  $P'$ .

*Proof:* Let  $Q$  be a quadrilateralization of  $P$ , and  $G(v, e)$  be a dual graph of  $Q$ . Now choose edge  $e_0 \in e$  which separates off a minimum number of nodes, that is at least 2. Let  $q \geq 2$  be this minimum, and  $v_0 \in v$  be the end node of link  $e_0$  in the separated part. The degree of nodes adjacent to  $v_0$  should be less than 2, because if degree of those nodes is greater than 1, then a cut through  $e_0$  is not a minimum cut. Therefore, the separated part has at most 4 nodes (i.e. 4 quadrilaterals). The remaining part of  $G$  is a tree with maximum degree 4, and it can be separated off again. Based on the number of nodes of  $G$ , one node may remain at the end of the cutting process. ■

The sufficient number of static guards to cover an orthogonal polygon, which is  $\lfloor \frac{n}{4} \rfloor$ , is based on assigning a guard to each pair of quadrilaterals. In order to track an evader inside the orthogonal environment, we assign a mobile guard to more than two quadrilaterals, and as a consequence we will reduce the number of static guards. We will proceed by partitioning  $P$  into smaller partitions, such that each partition contains  $q = 2, 3$  or 4 quadrilaterals. We will show that for each case, only a mobile guard is enough to track an evader for infinite time within a partition which contains  $q$  quadrilaterals. First, we show that assigning a mobile guard to each partition will result in fewer than  $\lfloor \frac{n}{4} \rfloor$  mobile guards.

**Lemma 7:** Orthogonal polygon  $P$  can be guarded by less than  $\lfloor \frac{n}{4} \rfloor$  mobile guards if each partition  $P_i$  can be guarded by one mobile guard, and  $P_i$ 's included at least two partitions with  $q = 3$ , or one partition with  $q = 4$ .

*Proof:* According to lemma 6, any orthogonal polygon  $P$  can be partitioned into parts which contain only 2, 3, or 4 quadrilaterals. Let  $n_2, n_3, n_4$  be the number of partitions with 2, 3, 4 quadrilaterals, respectively.  $k' \in \{0, 1\}$  corresponds to

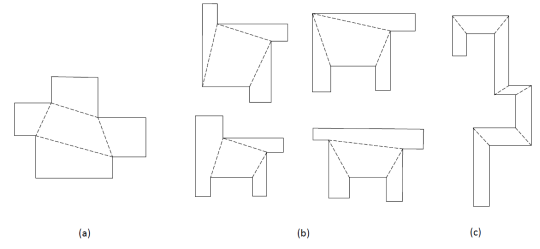


Fig. 8: (a) The only configuration for a degree 4 quadrilateral. (b) The possible configurations for a degree 3 quadrilateral. (c) The possible configuration for a degree 2 quadrilateral

the last quadrilateral  $P'$ . We will show  $n_2 + n_3 + n_4 + k' < \lfloor \frac{n}{4} \rfloor = \lfloor \frac{r+1}{2} \rfloor$ , where  $r$  is the total number of partitions.

$$r = 2n_2 + 3n_3 + 4n_4 + k', \quad (8)$$

$$\frac{r+1}{2} = n_2 + n_3 + n_4 + \frac{k'+1}{2} + \frac{n_3+2n_4}{2}, \quad (9)$$

$$\lfloor \frac{r+1}{2} \rfloor = n_2 + n_3 + n_4 + k' + \lfloor \frac{n_3+2n_4}{2} \rfloor. \quad (10)$$

For  $n_3 \geq 2$  or  $n_4 \geq 1$ ,  $\lfloor \frac{n_3+2n_4}{2} \rfloor \geq 1$ , lemma holds. ■

When all  $P_i$ 's contain only two quadrilaterals, that is the dual graph  $G$  is a line graph, we can take each pair of  $P_i$ 's and assign one mobile guard to them instead of assigning two static guards to them and reduce the required number of guards.

In case I, we assume that diagonal  $d$  separates off  $P_1$ , which contains 4 convex quadrilaterals. Without loss of generality, assume  $d$  is a diagonal  $AB$  in quadrilateral  $ABCD$ . Quadrilateral  $ABCD$  corresponds to a node with degree 4 in  $G(v, e)$ . Reflex vertices in a quadrilateral of degree 4 can have only one configuration, which is depicted in figure 8(a). Since both edges adjacent to  $d$ , which belong to  $P_1$ , should have the same orientation (i.e. both should be horizontal or vertical), both end points of  $d$  cannot be reflex vertices in polygon  $P_1$  simultaneously. Therefore, polygon  $P_1$  has three reflex vertices, two of which are  $270^\circ (3\pi/2)$ . Hence, two star regions should be quadrants and the connecting line between these reflex vertices is always inside one of the star regions. As a consequence, these star regions intersect, and the required maximum speed of guard ( $v_p^*$ ) can be found, based on the geometry of  $P_1$  and maximum speed of evader.

In case II, we assume that partition  $P_1$  contains 3 quadrilaterals, i.e.  $q = 3$ , and  $d$  belongs to a quadrilateral which corresponds to a node with degree 3 on graph  $G$ . A quadrilateral of degree three can have four configurations. These configurations are depicted in figure 8(b).  $P_1$  contains at most two reflex vertices in all possible configurations. Consequently, one mobile guard can track an evader within this part if the maximum speed of guard be greater than the required speed  $v_p^*$ . The necessary speed  $v_p^*$  can be characterized based on the geometry of star regions and maximum speed of evader  $\bar{v}_e$ .

When minimal partitioning  $P$  results in only 2 quadrilaterals, which implies that dual graph  $G$  is a line graph.

We can assign a mobile guard to each three consecutive quadrilaterals. In this case,  $P_1$  contains three quadrilaterals and each quadrilateral has a pair of horizontal or vertical edges. In this case,  $P_1$  contains only two disjoint star regions. Figure 8(c) shows such a condition.

## VI. CONCLUSIONS

In this work, we have proposed an algorithm to deploy a group of stationary and mobile guards in order to track an intruder inside a simply connected polygon. The environment is partitioned into smaller polygons, and a guard is assigned to each partition. Guards can be stationary or mobile depending on the geometry of the partition. The intruder is considered as a mobile agent with bounded speed. We provided a strategy to deploy stationary guards. For mobile guards, we presented an event-triggered strategy for their motion based on the intruder's trajectory in the dynamic zones. We have shown that the total number of guards is less than  $\lfloor \frac{n}{3} \rfloor$ , which is the sufficient number of point guards deployed for the coverage problem.

An ongoing effort in our research is to investigate cooperative task allocation among guards. The proposed algorithm is based on assigning a guard to each partition. Let  $G$  be a dual graph in which each node corresponds to a partition in  $P$ , and there an edge exists between the nodes if they share an edge in triangulation  $T$ . In the current algorithm, once the intruder enters the partition, the assigned guard tracks it independently. However, cooperation among guards which are adjacent in  $G$  will reduce the load on each guard. Another direction of the future research is to construct a road map for the guards which satisfies the necessary conditions of tracking. The current road map for the mobile guards is the line segment connecting the star regions or their intersections. This road map provides an upper bound for the maximum speed of the guard. However, the problem of finding the optimal speed for the guard can be cast as a convex optimization problem for certain subclasses of polygons, for example, polygons in which star regions are disjoint, and non-adjacent reflex vertices on the dual graph are not mutually visible. Other future directions of research are to incorporate sensing constraints, non-holonomic constraints and extension to 3 dimensional environments.

## REFERENCES

- [1] G. Viglietta, "Guarding and searching polyhedra," *arXiv preprint arXiv:1211.2483*, 2012.
- [2] C. Ding, A. A. Morfe, J. A. Farrell, and A. Roy-Chowdhury, "Coordinated sensing and tracking for mobile camera platforms," in *2012 American Control Conference (ACC)*. IEEE, 2012, pp. 5114–5119.
- [3] S. A. Quintero, F. Papi, D. J. Klein, L. Chisci, and J. P. Hespanha, "Optimal uav coordination for target tracking using dynamic programming," in *49th IEEE Conference on Decision and Control (CDC)*. IEEE, 2010, pp. 4541–4546.
- [4] E. Masehian and Y. Katebi, "Sensor-based motion planning of wheeled mobile robots in unknown dynamic environments," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 3-4, pp. 893–914, 2014.
- [5] A. K. Roy-Chowdhury and B. Song, "Camera networks: The acquisition and analysis of videos over wide areas," *Synthesis Lectures on Computer Vision*, vol. 3, no. 1, pp. 1–133, 2012.
- [6] R. Murrieta-Cid, T. Muppirla, A. Sarmiento, S. Bhattacharya, and S. Hutchinson, "Surveillance strategies for a pursuer with finite sensor range," *The International Journal of Robotics Research*, vol. 26, no. 3, pp. 233–253, 2007.
- [7] S. Bhattacharya and S. Hutchinson, "A cell decomposition approach to visibility-based pursuit evasion among obstacles," *The International Journal of Robotics Research*, vol. 30, no. 14, pp. 1709–1727, 2011.
- [8] L. J. Guibas, J.-C. Latombe, S. M. LaValle, D. Lin, and R. Motwani, "Visibility-based pursuit-evasion in a polygonal environment," in *Workshop on Algorithms and Data Structures*. Springer, 1997, pp. 17–30.
- [9] G. Laguna, R. Zou, and S. Bhattacharya, "Target tracking on triangulation graphs," in *International Conference on Intelligent Robots and Systems*, 2016, p. To Appear.
- [10] J. O'Rourke, *Art gallery theorems and algorithms*. Oxford University Press Oxford, 1987, vol. 57.
- [11] D. Lee and A. Lin, "Computational complexity of art gallery problems," *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 276–282, 1986.
- [12] V. Chvátal, "A combinatorial theorem in plane geometry," *Journal of Combinatorial Theory, Series B*, vol. 18, no. 1, pp. 39–41, 1975.
- [13] F. Hoffmann, M. Kaufmann, and K. Kriegel, "The art gallery theorem for polygons with holes," in *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*. IEEE, 1991, pp. 39–48.
- [14] D. Avis and G. T. Toussaint, "An optimal algorithm for determining the visibility of a polygon from an edge," *IEEE Transactions on Computers*, vol. 100, no. 12, pp. 910–914, 1981.
- [15] J. O'Rourke, "Galleries need fewer mobile guards: a variation on chvátal's theorem," *Geometriae Dedicata*, vol. 14, no. 3, pp. 273–283, 1983.
- [16] P. Tokekar and V. Isler, "Polygon guarding with orientation," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1014–1019.
- [17] M. J. Katz and G. Morgenstern, "Guarding orthogonal art galleries with sliding cameras," *International Journal of Computational Geometry & Applications*, vol. 21, no. 02, pp. 241–250, 2011.
- [18] T. Biedl, T. M. Chan, S. Lee, S. Mehrabi, F. Montecchiani, and H. Vosoughpour, "On guarding orthogonal polygons with sliding cameras," *arXiv preprint arXiv:1604.07099*, 2016.
- [19] M. Ghodsi, S. S. Mahdavi, and A. N. Sheshkalani, "Clearing an orthogonal polygon using sliding robots," *arXiv preprint arXiv:1607.03039*, 2016.
- [20] B. Tovar and S. M. LaValle, "Visibility-based pursuit-evasion with bounded speed," *The International Journal of Robotics Research*, vol. 27, no. 11-12, pp. 1350–1360, 2008.
- [21] K. J. Obermeyer, A. Ganguli, and F. Bullo, "A complete algorithm for searchlight scheduling," *International Journal of Computational Geometry & Applications*, vol. 21, no. 01, pp. 101–130, 2011.
- [22] A. Ganguli, J. Cortes, and F. Bullo, "Distributed deployment of asynchronous guards in art galleries," in *2006 American Control Conference*. IEEE, 2006, pp. 6–pp.
- [23] K. J. Obermeyer, A. Ganguli, and F. Bullo, "Multi-agent deployment for visibility coverage in polygonal environments with holes," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 12, pp. 1467–1492, 2011.
- [24] S. K. Ghosh, *Visibility algorithms in the plane*. Cambridge University Press, 2007.
- [25] S. M. LaValle, H. H. González-Banos, C. Becker, and J.-C. Latombe, "Motion strategies for maintaining visibility of a moving target," in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 1. IEEE, 1997, pp. 731–736.
- [26] C. Nam and D. A. Shell, "Assignment algorithms for modeling resource contention and interference in multi-robot task-allocation," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2158–2163.
- [27] K. Lerman, C. Jones, A. Galstyan, and M. J. Matarić, "Analysis of dynamic task allocation in multi-robot systems," *The International Journal of Robotics Research*, vol. 25, no. 3, pp. 225–241, 2006.